Evolution-based automated reconfiguration of field programmable analog devices

Adrian Stoica, Xin Guo*, Ricardo S. Zebulum, M. I. Ferguson, Didier Keymeulen
Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA 91109
* Chromatech, Alameda, CA 94501
adrian.stoica@jpl.nasa.gov

Abstract

The paper presents some experiments in automatic reconfiguration of field programmable devices using evolutionary algorithms. The experiments use a Field Programmable Transistor Array (FPTA), reconfigurable at transistor level. While parasitic effects of imperfect switches interconnecting transistors deteriorate performance of conventional designs when mapped to the FPTA, evolutionary algorithms are able to find a working design solution. The experiments are performed with a stand-alone board-level evolvable system (SABLES) in which the evolutionary algorithm is implemented with a DSP. SABLES can automatically configure the FPTA in tens to hundreds of seconds, time in which evaluates ~100,000 circuit candidate solutions. The paper overviews some examples of evolutionary synthesis of analog circuits on the FPTA.

1. Introduction

It is often heard that Field Programmable Devices offer the performance of hardware with the flexibility of software. In particular, beyond the flexibility of changing the system when needed (for prototyping, field upgrades, e.g. at a change of standard, to fix a bug, or in general at a change of specification) field programmable technology is now seen as a key contender for digital signal processing signal applications, in which the potential parallelism of programmable devices could offer significant computational speed. Furthermore, reconfigurable computing assumes the machine continuously morphing to best satisfy the particular computation to be performed, requirements and resources, and promises a true adaptive computing platform.

Unfortunately, the progress in the analog domain far lags behind the advances in the digital domain. There are several reasons for this, but key is the lack of suitable devices and computer assisted/ automated design tools. Imperfect interconnecting switches have a strong impact on the performance of an analog circuit and may badly deteriorate its response. Thus, taking a conventional design and mapping it on a programmable structure is not as straightforward matter as it is in digital – more likely such a design will be a poor one, except for simple cases. Thus, imperfect switches must be part of the design for the field-programmable analog device, at all stages including place and route. This appears a daunting problem.

A possible solution may come from the field of evolvable hardware (EHW), a recent research field, at the confluence of reconfigurable hardware, automated design and artificial intelligence. One could define evolvable hardware as hardware that reconfigures with the help of evolutionary algorithms. Automated hardware design using evolutionary algorithms involve the search for a candidate circuit design by a guided trial and error, performing the trials through consequent reconfigurations of the physical device. Thus, millions and possible billions of candidate designs are mapped into the reconfigurable hardware and their result tested against specifications, those who come closer getting a higher ranking. Unfortunately, as we discuss elsewhere, specifications, and complete testing (complete not only in going beyond functionality and including speed, power, etc, but also at different operational regions) of such circuits is very hard [1].

The evolutionary/genetic search in EHW is tightly coupled with a coded representation that associates each circuit to a "genetic code" or chromosome. The simplest representation of a chromosome is a binary string, a succession of 0s and 1s that encode a circuit. First, a population of chromosomes is randomly generated. The chromosomes are converted into bitstrings that configure a programmable device. The responses from the device are compared against specifications, and individuals are ranked based on how close they come to satisfying them. In preparation for a new iteration, a new population of individuals is generated from the pool of best individuals in the previous generation. This is subject to a probabilistic selection of individuals from a best individual pool,

0-7803-7574-2/02/$17.00 ©2002 IEEE

followed by two operations: random swapping of parts of their chromosomes, the *crossover* operation, and random flipping of chromosome bits, the *mutation* operation. The process is repeated for several generations, resulting in increasingly better individuals. Randomness helps to avoid getting trapped in local optima. Monotonic convergence (in a loose Pareto sense) can be forced by unaltered transference to the next generation of the best individual from the previous generation. There is no theoretical guarantee that the global optimum will be reached in a useful amount of time; however, the evolutionary / genetic search is considered by many to be the best choice for very large, highly unknown search spaces. The search process is usually stopped after a number of generations, or when closeness to the target response has reached a sufficient degree. One or several solutions may be found among the individuals of the last generation.

The path that EHW proposes to programmable analog is this: build/configure devices and perform a "Design by testing" on them. Testing a circuit directly in hardware avoids the problem of a design that needs to take in consideration models of the switches. The effect of the switches is there when the performance of a design is evaluated: if its response is correct than the problem is solved. This avoids also other problems related to the accuracy and drift of analog components, etc. What you tested is what you have. If components drift, a new evolution will find a new configuration that matches the specifications.

In order to test this hypothesis we built a few prototype programmable devices reconfigurable at transistor level. An evolutionary algorithm tested candidate designs on Field Programmable Transistor Arrays (FPTA). Through these experiments we learn about the possibilities and the difficulties of the EHW technique.

This paper briefly overviews the results of such experiments. A JPL FPTA chip acting as transistor-level reconfigurable hardware, is paired with a TI DSP implementing the evolutionary algorithm as the controller for reconfiguration. The solution called SABLES (Stand-Alone Board Level Evolvable System) allows fast (about 1,000 circuit evaluations per second) on-chip reconfiguration and testing of configured circuits. It achieves approximately 1-2 orders of magnitude reduction in memory and about 4 orders of magnitude improvement in speed compared to systems evolving in simulations, and about 1 order of magnitude reduction in volume and 1 order of magnitude improvement in speed (through improved communication) compared to a PC controlled system using the same FPTA chips. Details of SABLES appeared in [1].

The reminder of the paper overviews the FPTA, the integrated SABLES and illustrates automated reconfiguration of programmable devices in automated synthesis of a half-wave rectifier circuit and of a reconfigurable analog filter.

2. Examples of evolutionary reconfiguration of FPTA

The FPTA has transistor level reconfigurability, and supports any arrangement of programming bits without danger of damage to the chip. Three generations of FPTA chips have been built and used in evolutionary experiments. The FPTA concept is based on an array of transistors interconnected by switches. The latest chip, FPTA-2, consists of an 8x8 array of reconfigurable cells. Each cell has a transistor array as well as a set of other programmable resources, including programmable resistors and static capacitors. Figure 1 provides a broad view of the chip architecture together with a detailed view of the reconfigurable transistor array cell (another version of the cell is shown in Figure 3). The reconfigurable circuitry consists also of transistors connected through switches and is able to implement different building blocks for analog processing, such as two- and three-stage OpAmps, logarithmic photo detectors, or Gaussian computational circuits. It also includes capacitors and programmable resistors. Details of the FPTA can be found in [2,3].

The evolutionary algorithm was implemented in a DSP that directly controlled the FPTA, together forming a board-level evolvable system with fast internal communication ensured by a 32-bit bus operating at 7.5MHz. Details of the EP were presented in [4]. Over four orders of magnitude speed-up of evolution was obtained on the FPTA chip compared to SPICE simulations on a Pentium processor (this performance figure was obtained for a circuit with approximately 100 transistors; the speed-up advantage increases with the size of the circuit). The evaluation time depends on the tests performed on the circuit. Many of the evaluation tests performed required less than two milliseconds per individual, which for example on a population of 100 individuals running for 200 generations required only 20 seconds. The bottleneck is now related to the complexity of the circuit and its intrinsic response time. SABLES fits in a box 8" x 8" x 3".

The following experiment illustrates an evolution on SABLES. The objective of this experiment is to synthesize a half-wave rectifier circuit. The testing of candidate circuits is made for an excitation input of 2kHz sine wave of amplitude 2V. A computed rectified waveform of this signal is considered as the target. The fitness function rewards those individuals exhibiting behavior closer to target (using a simple sum of differences between the response of a circuit and target) and penalizes those farther from it. After evaluation of 100 individuals, they are sorted according to fitness and a 9% portion (elite percentage) is set aside, the remaining individuals undergoing first crossover (70% rate), either among themselves or with an individual from elite, and then mutation (4% rate). The entire population is then reevaluated. In this experiment the search of a configuration was limited to using only two FPTA cells.

Figure 2, on the left side, displays snapshots of evolution in progress, illustrating the response of the best individual in the population over a set of generations. The first caption shows the best individual of the initial population, while the subsequent ones show the best after 5, 50 and 82 generations. The solution, with a fitness below 4,500 is shown on the right.

The operational range of the evolved circuit in the frequency domain is a possible concern, since in principle the circuit behavior should be evaluated for the overall frequency domain in which it is expected to work. While the circuit works properly for the decade going from 500Hz to 5kHz, the response deteriorates for higher frequencies, e.g. 50kHz. These results are typical of a series of successful runs. Approximately 1 out of 10 runs ended with the algorithm getting stuck and not finding a solution at all for that small/fixed mutation rate.

Another example is that of a reconfigurable filter, shown in Figure 3 with several characteristics corresponding to various configuration bitstrings. Filters are in fact the main application of Field Programmable Analog Arrays (FPAA) at this time. One should point out however that this is a different type of filter, largely changing because of the characteristic of the transistor circuit itself (as a nonlinear circuit) as opposed to FPAA designs where switches interconnect linear Operational Amplifier; this has the potential to be a more compact solution (here only one FPTA cell). On the other hand on FPAA one can map conventionally designed filters. (The purpose here is to illustrate an approach of automated on-chip design by evolution, and not to claim improved performance over filters with FPAA – clearly that is currently an approach that is safer and with greater performance).

4. Conclusions

Experiments demonstrate automated reconfiguration of programmable devices using evolutionary algorithms. In particular this offers a solution to synthesis of analog circuits. This approach also demonstrates the possibility of analog reconfigurable computing using evolvable hardware. Evolutionary design moves the burden from design of a circuit (configuration) to design of a search technique (including here search parameters, but most importantly fitness functions), which is not necessarily a simpler thing. Still this is a promising path and further exploration is justified.

Acknowledgements

The research described in this paper was performed at the Center for Integrated Space Microsystems, Jet Propulsion Laboratory, California Institute of Technology and was sponsored by the Defense Advanced Research Projects Agency (DARPA) and the National Aeronautics and Space Administration (NASA).

References

[1] A. Stoica, R. Zebulum, M.I. Ferguson, D. Keymeulen, V. Duong and Xin Guo, "Evolving Circuits in Seconds: Experiments with a Stand-Alone Board Level Evolvable System, Proc of 2002 NASA/DoD Conference on Evolvable Hardware, Arlington, VA July 15-18, 2002, IEEE Computer Society Press, pp. 67-74

[2] A. Stoica, R. Zebulum, D. Keymeulen, R. Tawel, T. Daud, and A. Thakoor, Reconfigurable VLSI Architectures for Evolvable Hardware: from Experimental Field Programmable Transistor Arrays to Evolution-Oriented Chips. In IEEE Transactions on VLSI Systems, Special Issue on Reconfigurable and Adaptive VLSI Systems, vol. 9, No. 1,February 2001. (pp.227-232).

[3] A. Stoica, D. Keymeulen, A. Thakoor, T. Daud, G. Klimech, Y. Jin, R. Tawel, V. Duong. Evolution of Analog Circuits on field Programmable Transistor Arrays. In J Lohn et al.(eds.),Proceedings of 2nd NASA/DoD Workshop on Evolvable Hardware (EH2000), July 13-15,2000, (pp.99-108). Palo Alto, CA, USA. IEEE Computer Society.

[4] Ferguson, M.I., Stoica A., Zebulum R., Keymeulen D. and Duong, V. "An Evolvable Hardware Platform based on DSP and FPTA". Proceedings of the Genetic and Evolutionary Computation Conference,(GECCO) July 9-13, 2002, New York

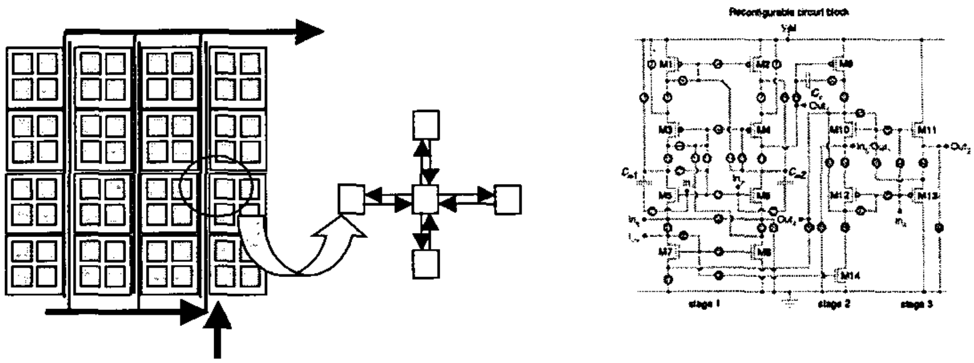

Figure 1. FPTA 2 architecture (left) and schematic of cell transistor array (right). The cell contains additional capacitors and programmable resistors (not shown).

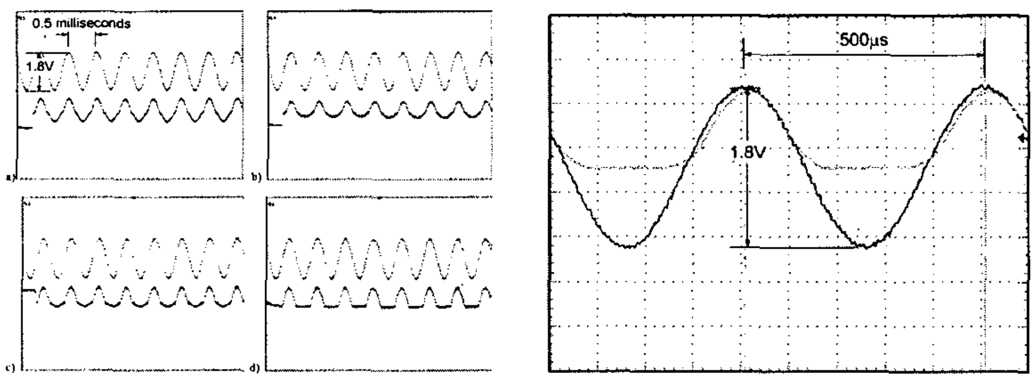

Figure 2. Evolution of a halfwave rectifier showing the response of the best individual of generation a) 1, b) 5, c) 50 and finally the solution at generation d) 82. The final solution is magnified on the right.

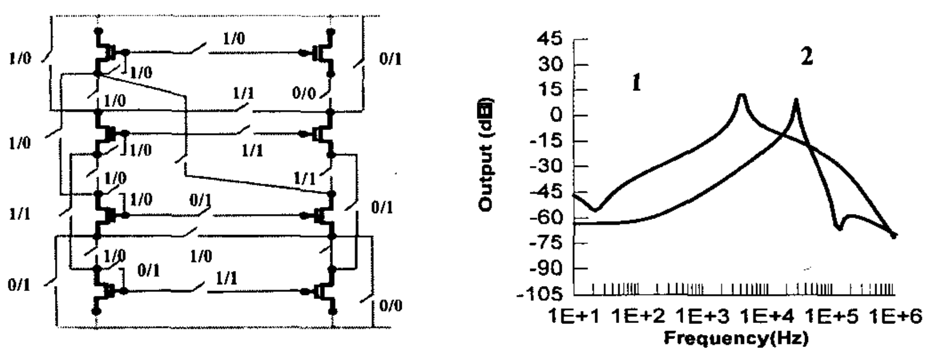

Figure 3. Filter on re-configurable circuit. Simplified architecture at the left and response at the right. (capacitors omitted in the figure). The binary state of the switches is represented next to the respective switch for the two filters. Filter 1 presents a gain of 11dB at 5kHz and roll-off about −30dB/dec. Filter 2 presents a gain of 9dB at 25kHz and roll-off about −40dB/dec for the lower band and −70dB/dec for the upper band.